Learning 3D Appearance Models from Video

Fernando De la Torre † **Jordi Casoliva** † **Jeffrey F. Cohn** ‡
ftorre@cs.cmu.edu casoliva@cs.cmu.edu jeffcohn@pitt.edu

†Robotics Institute, Carnegie Mellon University,
Pittsburgh, Pennsylvania 15213.

‡Department of Psychology, University of Pittsburgh,
Pittsburgh, Pennsylvania 15260.

Abstract

Within the past few years, there has been a great interest in face modeling for analysis (e.g. facial expression recognition) and synthesis (e.g. virtual avatars). Two primary approaches are appearance models (AM) and structure from motion (SFM). While extensively studied, both approaches have limitations. We introduce a semi-automatic method for 3D facial appearance modeling from video that addresses previous problems. Four main novelties are proposed:

- *A 3D generative facial appearance model integrates both structure and appearance.*
- *The model is learned in a semi-unsupervised manner from video sequences, greatly reducing the need for tedious manual pre-processing.*
- *A constrained flow-based stochastic sampling technique improves specificity in the learning process.*
- *In the appearance learning step, we automatically select the most representative images from the sequence. By doing so, we avoid biasing the linear model, speed up processing and enable more tractable computations.*

Preliminary experiments of learning 3D facial appearance models from video are reported.

1 Introduction

Within the past few years, there has been great interest in face modeling for analysis (e.g. facial expression recognition) and synthesis (e.g. virtual avatars). Among various approaches to modeling 3D faces from video, two of the most popular and commonly used are based on appearance models (AM) [2, 4, 8, 9, 17] and rigid/nonrigid structure from motion (SFM) [5, 7, 16, 21]. While each has been studied extensively, both approaches suffer from several drawbacks. All SFM approaches have an implicit data conservation assumption in their formulation, since the correspondence problem is usually solved with classical trackers or flow techniques. In the face domain, this aspect is dramatic as the face undergoes deep changes in appearance due to variations in expression that may be either qualitative (e.g. blinking, appearance of the tongue, etc) as well as quantitative (i.e., intensity change), which can seriously bias any parameter estimation.

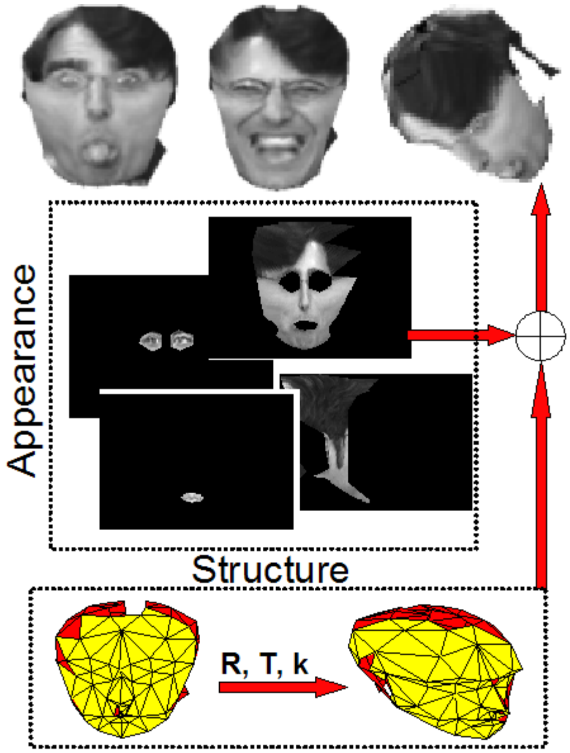

Figure 1: Generative 3D Facial Appearance Model with Structure and Appearance.

While the AM approach overcomes the problem of appearance change by explicitly introducing linear variation of intensity and shape, it incurs other challenges. AM approaches do not necessarily decouple the rigid/non-rigid motion in the fitting process, since a single shape basis models both of them. Moreover, AM approaches require a labeled training set to learn face appearance. Manual labeling of face images is tedious and prone to error. In this paper, we propose a generative model that is robust to intensity changes in appearance, takes into account structure and appearance, and learns model parameters in a semi-supervised manner. Fig. 1 illustrates the main idea of the paper.

2 Previous Work

It is beyond the scope of the paper to review all the work related to 3D face modeling. Notwithstanding, we cite the more relevant literature. Several papers have used a relatively simple 3D model (e.g. cylinder [19], ellipse, etc) and flow equations to recover 3D rigid head motion. These approaches model 3D rigid motion but only crudely the 3D shape of the face (relative to SFM and AM).

In the area of structure from motion (SFM), several authors have reported encouraging results. Torresani et al. [18] decouple rigid and non-rigid motion under orthographic projection. Chowdhury and Chellappa [7] construct a 3D model by inferring depth from flow. In a similar approach but using feature correspondences and performing bundle adjustment, Zhang et al. [21] construct 3D models from a video in which the face rotates 180 degrees from profile to profile. Pighin et al. [16] model and animate 3D Face Models using SFM in multi-view images and solving the correspondence by hand. Brand [5] reports a SFM technique using a new algebraic approach that allows accommodation for uncertainty and is less prone to propagating errors.

Since active shape model/active appearance models [8] and Morphable models [13] appeared, there has been much related work in the appearance/face domain. Vetter and Blanz [4] have introduced morphable models learned from a Cyberscan, which takes into account shape and texture. Romdhani and Vetter [17] have recently improved the fitting process (see [1] for efficient fitting). Black and Jepson [3] introduce an elegant formulation for continuous alignment w.r.t. a subspace. Cascia et. al. [6] show a method that is able to track 3D heads under changeable illumination conditions by registering w.r.t the eigenspace. While the AM approach has shown great performance, the various algorithms require training from hand-labeled samples, which is labor intensive and error prone.

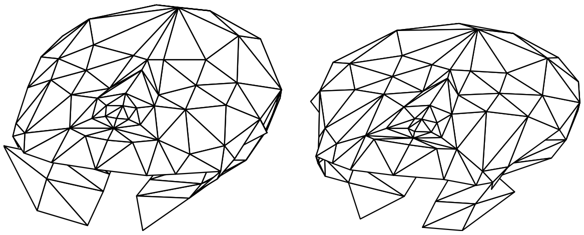

Figure 2: a) Original 3D Mesh. b)Deformed 3D Mesh.

Frey and Jojic [11] introduced an Expectation Maximization (EM) algorithm that learns several statistical models (e.g. PCA, mixture of Gaussians, etc) that are invariant to geometric transformations. However the complexity of the algorithm scales exponentially with the number of motion parameters. To solve this problem, De la Torre and Black [10] proposed an energy function based algorithm to learn the appearance model. Their algorithm achieves invariance to geometric transformation while remaining scalable. In related but independent work, Baker et al. [2] have proposed a method to learn the AAM in a unsupervised fashion. Morency et al. [15] introduce an adaptive view-based appearance model, which is able to register w.r.t. previous selected prototypes. The method we present in this paper benefits from previous AM and SFM approaches, by learning a structured appearance model in an unsupervised manner.

3 Generative Model for 3D Faces

In this section we describe a possible generative 3D facial appearance model that takes into account the structure, appearance and 3D motion.

3.1 From Generic 3D Structure to Person-Specific Models

We begin with a generic 3D head model (http://grail.cs.washington.edu/projects/realface/) and subsample it to make it more computationally tractable. To give a first estimation of the shape of the face, we select 30 points by hand in two orthogonal views. The mesh is then deformed using a radial basis function and affine transformation that minimizes: $E(\mathbf{C},\mathbf{A}) = ||\mathbf{P}_{2d} - \mathbf{C}\mathbf{D} - \mathbf{A}\mathbf{P}_{3d}||_F$ subject to $\mathbf{C}^T\mathbf{1} = \mathbf{0}$ and $\mathbf{C}^T\mathbf{P}_{3d} = \mathbf{0}$, where $\mathbf{P}_{2d} = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{pmatrix}$ are the 2D image points, $\mathbf{P}_{3d} = \begin{pmatrix} X_1 & X_2 & \cdots & X_n \\ Y_1 & Y_2 & \cdots & Y_n \\ 1 & 1 & \cdots & 1 \end{pmatrix}$ are the 3D points of the mesh. $\mathbf{A} \in \Re^{2\times 3}$ contains an affine transformation, $\mathbf{D}$ is a matrix such that each element $d_{ij} = exp^{-\frac{(X_i-X_j)^2+(Yi-Yj)^2}{\beta}}$ is the Euclidian distance [16]. Once we have re-escaled the X,Y axis, we do a

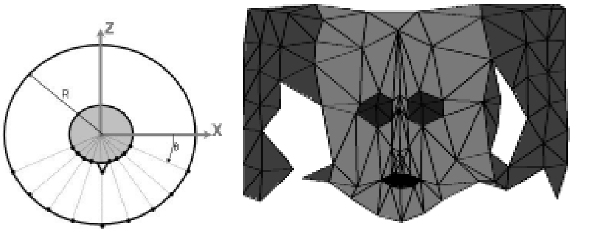

Figure 3: a)Projection into cylindrical coordinates. b)Unwarped Mesh.

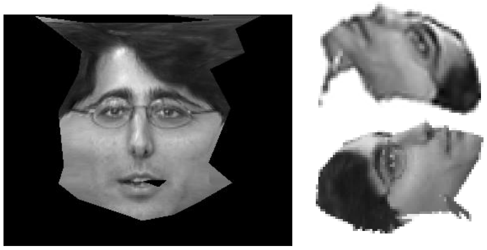

Figure 4: a)Texture mapped from one image to the unwarped cylinder. b)Two views of the texture map in 3D.

similar approach to re-scale the Z axis. In fig. 2.a, we see the original 3D mesh and in fig. 2.b we can see the person-specific model once deformed.

3.2 Modeling Appearance Changes

Once the structure of the face is obtained, we construct the appearance model by mapping the 3D model into cylindrical coordinates. In figure 3.a it is possible to see how to project the mesh into cylindrical coordinates, y = Y and x = arctan(αX/Z) where α is a variable which adjust the cylindrical projection. In figure 3.b we can see the unwarped mesh.

Once we have unwarped the mesh, we map the texture from the image to the unwarped mesh, assuming perspective projection. Similar to previous work [10], in the unwarped texture image, we define four regions in the unwarped texture image, corresponding to the eyes, mouth, profiles and the rest of the face. Each of the regions contains a subspace of different dimensionality (figure 4.a). After the unwarped texture is obtained, it is mapped from the unwarped cylindrical parameter space to the 3D model, by means of the triangular patches [16] (figure 4.b).

4 Flow based initialization

We use flow based techniques to give an initial and fast estimation of the rotational and translational components of the rigid motion of the head between frames[19]. However, flow techniques are based on the brightness constancy assumption and are well known for being noisy and ambiguous when recovering 3D information. To overcome these difficulties, we make use of robust statistics techniques [3] and approximate the average head depth with a simple 3D model. Candidate 3D models include cylindrical [19], ellipsoidal and anthropomorphic models. Within a coarse-to-fine iterative strategy, we minimize[1]:

$$E(\boldsymbol{\mu}) = \sum_{p \in R} \rho(d_{pt}(\mathbf{f}(\mathbf{x}_p, \boldsymbol{\mu})) - d_{p(t-1)}(\mathbf{f}(\mathbf{x}_p, \mathbf{0})), \sigma) \quad (1)$$

where $\rho(x, \sigma) = \frac{x^2}{x^2+\sigma^2}$, $\boldsymbol{\mu} = (\dot{\boldsymbol{\theta}}, \mathbf{t}) = (\theta_x, \theta_y, \theta_z, t_x, t_y, t_z)$ are the parameters for the rotational and translation components and R is the region of support. $\mathbf{f}(\mathbf{x}, \boldsymbol{\mu})$ is the geometric transformation:

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\mu}) = \begin{pmatrix} f_x \frac{\mathbf{R}(\theta_x,\theta_y,\theta_z)(X \ Y \ Z) + t_x}{\mathbf{R}(\theta_x,\theta_y,\theta_z)(X \ Y \ Z) + t_z} - x_0 \\ f_y \frac{\mathbf{R}(\theta_x,\theta_y,\theta_z)(X \ Y \ Z) + t_y}{\mathbf{R}(\theta_x,\theta_y,\theta_z)(X \ Y \ Z) + t_z} - y_0 \end{pmatrix}$$

where $\mathbf{R}(\theta_x, \theta_y, \theta_z)$ is a rotation matrix and $[X \ Y \ Z]^T$ are the 3D coordinates (the intrinsic camera parameters are known).

Minimizing expression (1) becomes a non-linear estimation problem due to the robust function and the behavior of the motion parameters. To approximate the problem by a linear one, we linearize the motion variation and use the Iteratively Reweighted Least Squares (IRLS) algorithm [14, 10]. Given an initial estimation of the motion parameters $\boldsymbol{\mu}^0$, a Gauss-Newton method can be applied by incrementally updating the parameters solving the following approximate minimization problem:

$$E(\boldsymbol{\mu}) \approx ||\mathbf{d}_t(\mathbf{f}(\mathbf{x}, \boldsymbol{\mu}^0)) + \mathbf{J_t} \boldsymbol{\Delta\mu} - \mathbf{d}_{t-1}(\mathbf{f}(\mathbf{x}, \mathbf{0}))||_{\mathbf{W}_t} \quad (2)$$

where $\mathbf{J_t} = \frac{\partial \mathbf{d_t}}{\partial \boldsymbol{\mu}}$ is the Jacobian matrix. See [10, 19].

5 Dimensionality Reduction

Dimensionality reduction is a common technique to filter and makes algorithms more computationally tractable. When processing large videos of the same person, the amount of redundant facial expression/poses becomes an issue for several reasons.

[1] Throughout this paper, we will use the following notation: bold capital letters denote a matrix $\mathbf{D}$, bold lower-case letters a column vector $\mathbf{d}$. $\mathbf{d}_j$ represents the j-th column of the matrix $\mathbf{D}$. dij denotes the scalar in the row i and column j of the matrix $\mathbf{D}$ and the scalar i-th element of a column vector $\mathbf{d}_j$. All nonbold letters will represent scalar variables. $||\mathbf{d}||^2_{\mathbf{W}} = \mathbf{d}^T\mathbf{W}\mathbf{d}$ is a weighted norm of a vector $\mathbf{d}$. diag is an operator which transforms a vector to a diagonal matrix. $\mathbf{D}_1 \circ \mathbf{D}_2$ denotes the Hadamard (point wise) product between two matrices/vectors of equal dimensions.

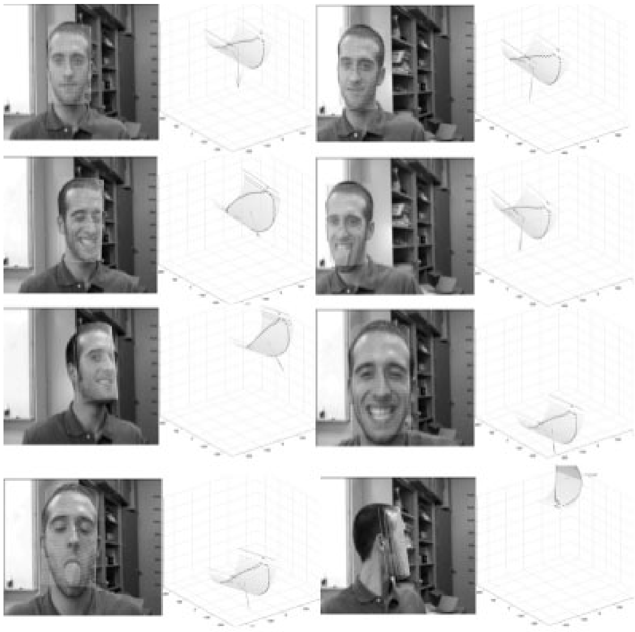

Figure 5: a) Example of tracking results with pose and facial expression changes.

Firstly, we do not necessarily have a uniform sampling of all the possible facial expressions/poses. This will bias the appearance learning algorithm towards reconstructing better the expressions with more samples. Secondly and more importantly, the amount of data would make the stochastic algorithm very computationally expensive. To avoid this phenomena, once the images are registered, we find the most representative prototypes by clustering, using the recent advances in multi-way normalized cuts [20]. In figure (6) we show 50 prototypes extracted from a sequences of 800 frames. Figure (7) shows some of the samples of the same cluster. We can observe that individual prototypes capture changes in expression/pose.

6 Stochastic Smoothing for Appearance Learning

The optical flow provides a first estimation of the rigid motion parameters, which can be biased due to changes in facial expression, the fact that the 3D model is not accurate enought and linealization errors. In order to improve the estimation, compute non-rigid motion parameters, and build the appearance model, we use a smoothing particle filtering algorithm [12]. We pose the problem as doing inference in a general state space model, which can be described by: $\mathbf{s}_t = g(\mathbf{s}_{t-1}, \mathbf{u}_t) + \beta_t$ and $\mathbf{d}_t = h(\mathbf{s}_t) + \xi_t$ where $\mathbf{d}_t$ is the vectorized observed image frame at time t. The hidden state, $\mathbf{s}_t$, will recover $(\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \kappa)$, where κ are the non-rigid parameters (see section 6.1). $\mathbf{u}_t$ is the input and β_t and ξ_t are samples from a noise distribution. h is the measurement function and g describes the dynamics of the system.

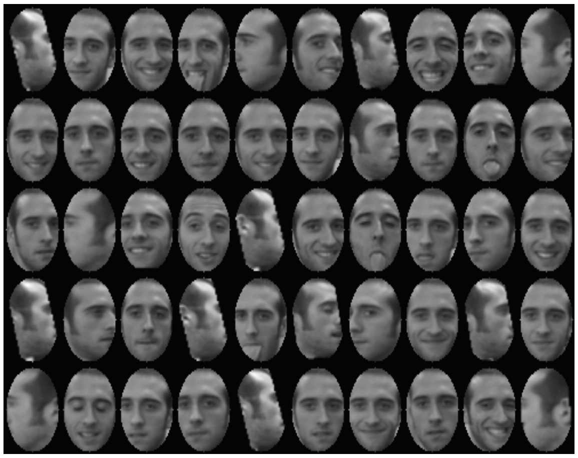

Figure 6: 50 prototypes extracted from 800 frames.

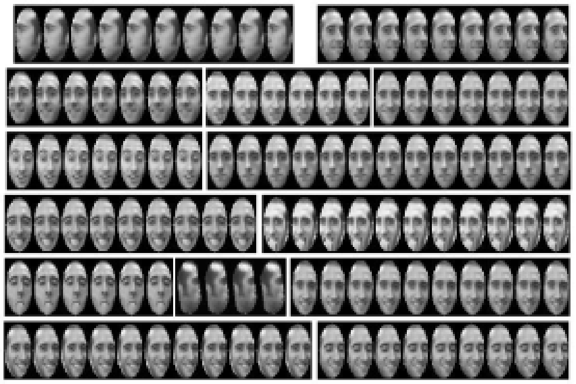

Figure 7: Samples of several clusters.

6.1 Measurement Equation

The measurement equation expresses the fact that an image at time t, $\mathbf{d}_t$, is generated by a general non-linear function h of $\mathbf{s}_t$. The likelihood of a particular sample of $\mathbf{s}_t$ is related to the image by:

$$M^* = NR(\mathbf{R}(\theta_x, \theta_y, \theta_z) \cdot \mathbf{M} + [t_x, t_y, t_z]^t, \kappa) \quad (3)$$

$$p(\mathbf{d}_t | \boldsymbol{\mu}, \kappa) \sim exp - \frac{||\mathbf{d}_t - Rec(\mathbf{d}_t(Proj(M^*)))||}{\sigma} \quad (4)$$

where we define several operators; $\mathbf{M} = \begin{pmatrix} X_1 & \cdots & X_n \\ Y_1 & \cdots & Y_n \\ Z_1 & \cdots & Z_n \end{pmatrix}$ is the centered 3D mesh. $NR(\mathbf{M}, \kappa)$ is an operator which takes the 3D mesh and deforms the non-rigid parameters κ. κ is a vector of 3 parameters which modify the positions

of eyebrows, mouth corners and the mandible aperture. $Proj$ is the perpective projection operator $[f_x X/Z - x_0, f_y Y/Z - y_0]$ of the visible triangles in the 3D mesh. Given the projected visible triangles, Rec takes the image triangles, projects them into cylindrical coordinates and reconstruct the subspace as $\sum_{l=1}^{L}(\boldsymbol{\pi}_t^l \circ \mathbf{B}^l \mathbf{c}_t^l)$, where:
$\pi_{\mathbf{t}}^{\mathbf{l}}$: Binary mask of the l layer at time t, which represents its spatial domain. $\boldsymbol{\pi}_t^l = [\pi_{1t}^l \ \pi_{2t}^l \cdots \pi_{dt}^l]$, where each $\pi_{pt}^l \in \{0,1\}$and $\sum_l \pi_{pt}^l = 1 \quad \forall p, t$. It is defined by hand.
$\mathbf{c}_t^l$: Coefficients which linear combination of the basis $\mathbf{B}^l$ will reconstruct the graylevel of the layer l.
$\mathbf{B}^l$: Appearance Basis of the l layer.
Observe that equation (4) represents a pseudolikelihood (not necesarily normalized).

6.2 State Equation

Eq. (4) describes the dynamical behavior of the hidden states of the dynamical system (the image sequence). In the more general case $g(\mathbf{s}_t, \mathbf{u}_t)$ is a nonlinear transformation (e.g. a mixture of gaussians, a multilayer perceptron network, $\cdots$).
The optical flow has given a first estimation of the 3D rigid parameters up to a scale factor due to the ambiguity between translation and depth. Despite the fact that the flow estimation can be a little bit biased, we use it to guide the search while sampling the posterior distribution of the state parameters. We combine both estimations(flow and temporal) with their covariances in an optimal Bayesian way:

$$p(\mathbf{s}_t|\mathbf{s}_{t-1}, \mathbf{f}_t) = N(\Sigma_t^{-1}(\Sigma_d^{-1}\mathbf{A}\mathbf{s}_{t-1} + \Sigma_f^{-1} \cdot \mathbf{f}_t), \Sigma_t) \quad (5)$$

$$\Sigma_t^{-1} = \Sigma_d^{-1} + \Sigma_f^{-1} \quad (6)$$

where Σ_d is the uncertainty comming from the dynamical system, $\mathbf{f}_t$ is flow estimation for the rigid parameters, Σ_t is the uncertainty of the computed optical flow. To compute an estimation of Σ_f, we run several iterations of Gauss-Newton with IRLS method, and, once it has converged, we recompute the Jacobian $\mathbf{J_t}$ with the final parameter values $\mathbf{f}_t$ and a binary weighting matrix $\mathbf{W}_t$ is constructed. Then, an estimation of the uncertainty is given by $\Sigma_t = trace(\mathbf{W}_t)(\mathbf{J_t^T W_t J_t})^{\mathbf{-1}}$. $\mathbf{A}$ stands for a simple linear dynamical model, which is assumed to have a constant velocity model. Once the parameters are known, we sample from the multidimensional gaussian to generate new samples.

6.3 Deterministic Gradient Learning

Having a reasonable assessment of the rigid/non-rigid parameters over a set of k frames, we unwarp the texture and compute an estimation of the subspace for each region of the face. For each unwarped frame, we have an image $\mathbf{p}_t \in \Re^{k_t \times 1}$ and a weighting matrix $\mathbf{w}_t \in \Re^{k_t \times 1}$. We minimize $E(\mathbf{B}^1, \mathbf{C}^1, \cdots, \mathbf{B}^l, \mathbf{C}^l) = ||\mathbf{W} \circ (\mathbf{P} - \sum_{l=1}^{L} \boldsymbol{\pi}^l \mathbf{B}^l \mathbf{C}^l)||_F$, where $\mathbf{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_k] \in \Re^{k_t \times k}$ is a matrix such that $w_{ij} = 1$ is the pixel which is visible and $w_{ij} = 0$ if not. $\mathbf{B}^l \in \Re^{d \times k}$ is the set of k basis and $\mathbf{C}^l = [\mathbf{c}_1^l \cdots \mathbf{c}_n^l] \in \Re^{k \times n}$ are the set of coefficients for the l layer. We recursively update the basis to preserve 85% of the energy. To optimize for $\mathbf{B}$ and $\mathbf{C}$, we use a two step method which alternates between minimizing $\mathbf{C}$ in closed form with $\mathbf{B}$ fix and viceversa until convergence. See [10] for more details.

7 Experiments

Figure 8 shows some pictures with the tracking results. The projected 3D mesh onto the images is shown as well as the original rotated 3D mesh. The original sequence has approximately 800 frames from which, after tracking with flow (section 5) and clustering, 130 frames are selected. From each of the 130 frames, we have taken subsets of 15 frames, compute an appearance basis and run the smoothing for Condensation in order to register w.r.t the subspace. We iterate the learning step and the smoothing algorithm until convergence. Good results have been achieved using 700 particles and, tipically, 3 runs going backward and forward for the smoothing process. The algorithm has been implemented in a non-optimized Matlab code and takes roughly 7 hours to process the original image sequence of 800 frames. This takes into account the Optical Flow, Condensation, smoothing for Condensation and the learning of the appearance model. Figure 9 shows the 3D mesh with the learned appearance model.

8 Acknowledgements

This work has been partially supported by National Business Center of the Department of the Interior under a subcontract from SRI International, U.S. Department of Defense contract N41756-03-C4024, NIMH Grant R01 MH51435 and DARPA HumanID program under ONR contract N00014-00-1-0915.

References

[1] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision, 2004*, 2004.

[2] S. Baker, I. Matthews, and J. Schneider. Image coding with active appearance models. Technical Report 03-13, CMU-RI, 2003.

[3] M. J. Black and A. D. Jepson. Eigentracking: Robust matching and tracking of objects using view-based representation. *International Journal of Computer Vision*, 26(1):63–84, 1998.

[4] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Siggraph 1999, Computer Graphics Proceedings*, pages 187–194, 1999.

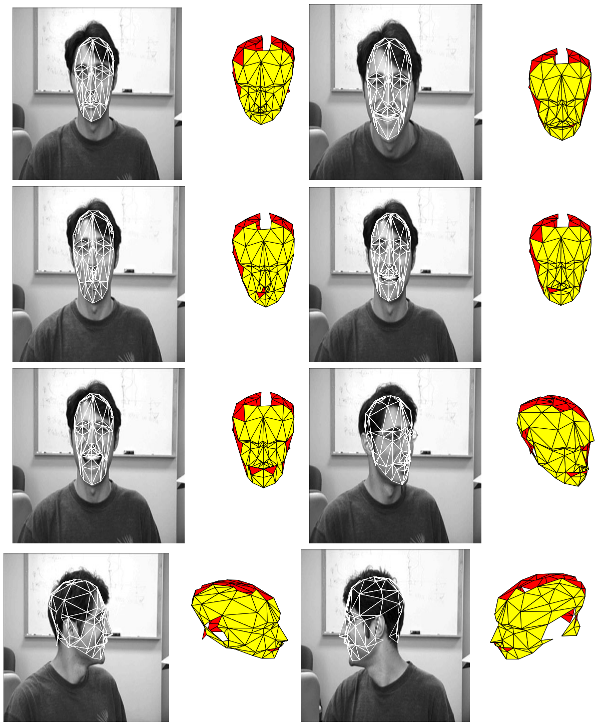

Figure 8: 3D tracking and the projected mesh.

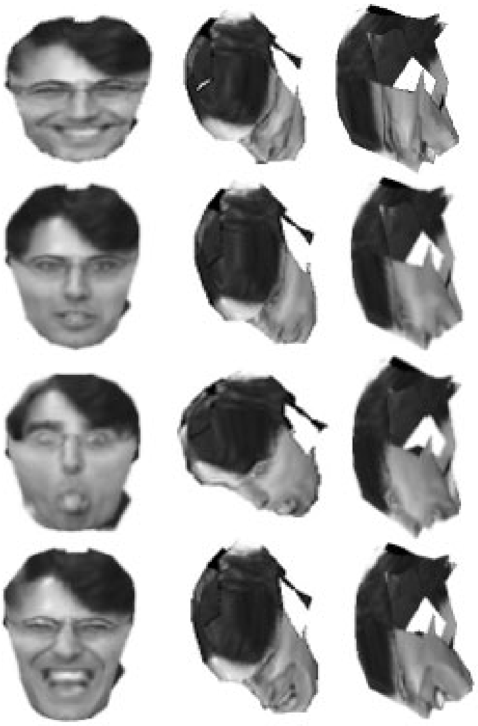

Figure 9: 3D mesh with the learned appearance model.

[5] M. Brand. 3d morphable models from video. In *Conference on Computer Vision and Pattern Recognition*, 2001.

[6] M. Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models, 1999.

[7] A. K. Chowdhury and R. Chellappa. Registration of partial 3D models extracted from multiple video streams. In *IEEE International Workshop on Multimedia and Signal Processing*, 2002.

[8] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *European Conference Computer Vision*, pages 484–498, 1998.

[9] F. de la Torre and M. J. Black. Robust parameterized component analysis: theory and applications to 2d facial appearance models. *Computer Vision and Image Understanding*, 91:53 – 71, 2003.

[10] F. de la Torre and M. J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 1(54):117–142, In press, 2002.

[11] B. J. Frey and N. Jojic. Estimating mixture models of images and inferring spatial transformations using the em algorithm. In *Conference on Computer Vision and Pattern Recognition*, pages 416–422, 1999.

[12] M. Isard and A. Blake. A smoothing filter for CONDENSATION. *European Conference on Computer Vision*, 1406, 1998.

[13] M. J. Jones and T. Poggio. Multidimensional morphable models. In *International Conference on Computer Vision*, pages 683–688, 1998.

[14] G. Li. Robust regression. In D. C. Hoaglin, F. Mosteller, and J. W. Tukey, editors, *Exploring Data, Tables, Trends and Shapes*. John Wiley & Sons, 1985.

[15] L. Morency, A. Rahimi, and T. Darrell. Adaptive view-based appearance model. In *CVPR*, 2003.

[16] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin. Synthesizing realistic facial expressions from photographs. *Computer Graphics*, 32(Annual Conference Series):75–84, 1998.

[17] S. Romdhani and T. Vetter. Efficient, robust and accurate fitting of a 3D morphable model. In *International Conference Computer Vision*, pages 59–66, 2003.

[18] L. Torresani, D. Yang, E. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints, 2001.

[19] J. Xiao, T. Kanade, and J. F. Cohn. Robust full-motion recovery of head by dynamic templates and re-registration techniques. In *IEEE International Conference on Automatic Face and Gesture Recognition*, May 2002.

[20] S. Yu and J. Shi. Multiclass spectral clustering. In *ICCV*, 2003.

[21] Z. Zhang, Z. Liu, D. Adler, M. Cohen, E. Hanson, and Y. Shan. Robust and rapid generation of animated faces from video images: A model-based modeling approach. Technical Report MSR-TR-01-101, Microsoft Research, October 2001.